DOCKET NO.: 05-03-010 **PATENT**
CLIENT NO.: UGSC01-05022
Customer No. 45113

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Application of	:	Graham Hughes, et al.
Serial No.	:	10/706,848
Filed	:	November 12, 2003
For	:	SYSTEM, METHOD, AND COMPUTER PROGRAM PRODUCT FOR DISTRIBUTED TESTING OF PROGRAM CODE
Group No.	:	2192
Examiner	:	Eric B. Kiss

MAIL STOP AF
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Sir:

PRE-APPEAL BRIEF REQUEST FOR REVIEW

Applicant requests review of the final rejection in the above-identified application. No amendments are being filed with this request.

This request is being filed with a notice of appeal. The review is requested for the reason(s) stated in the arguments below, demonstrating the clear legal and factual deficiency of the rejections of some or all claims. Applicant notes that the Examiner is correct that claim 8 is inadvertently missing a word. Applicant would be pleased to have this corrected by Examiner's amendment, or to

otherwise correct this error as would be satisfactory to the Examiner, or can proceed to appeal on the remaining claims.

Claims 1-21 were rejected under 35 U.S.C. § 103(a) as being unpatentable over U.S. Patent No. 6,112,225 to *Kraft, et al.*, hereinafter "Kraft" and U.S. Patent No. 6,360,268 to *Silva, et al.*, hereinafter "Silva". For the convenience of the Panel, claim 1 requires:

> 1. (Original) A method for testing code, comprising:
> receiving a test request;
> sending executable program code, corresponding to the test request, to a client system;
> receiving a response from the client system indicating that the client system will perform a test, and indicating that the client system was not being actively used when the executable program code was sent.

Claim 1 requires receiving a test request. The Examiner argues that this is taught a Kraft at col. 9, lines 1-27. This portion of Kraft was reproduced in a previous response, and of course does not teach or suggest receiving a test request at all. Kraft does teach that a subtask request is sent to coordinating computer 102 (which, of course, receives it).

Claim 1 also requires sending executable program code, corresponding to the test request, to a client system. Kraft appears to teach that a subtask is then received by peripheral computer 106, which may have been sent by the coordinating computer 102.

Claim 1 also requires receiving a response from the client system indicating that the client system will perform a test, and indicating that the client system was not being actively used when the executable program code was sent. This is <u>not</u> taught or suggested by Kraft. The Examiner refers again to the passage reproduced again, but it is clear that coordinating computer 102 does not receive

any sort of response indicating that the peripheral computer will perform a subtask, nor does coordinating computer 102 receive any sort of response indicating that the peripheral computer was not being actively used when the subtask was sent.

The Examiner argues that task requests and results are sent when the system is idle – but this does not meet the claim limitations. There is no response sent from the client to server indicating that the client will perform a task that it has been sent, as claimed – there appears to be no confirmation at all that a subtask has been received, or that the client will execute it. Nor is there any response sent from the client to server indicating that the client was not being actively used when the subtask was sent – it may have been idle when the task was originally requested, and it may be idle again at some point when the task is completed, but there is no indication that the client is idle when the subtask is received.

This can be important, of course, when the server must determine how soon a task is to be completed – in Kraft's system, there is no indication at all of when the task may be executed or completed. On the other hand, the claimed method provides that when the code is received by the client, it responds with an indication that it was idle on receipt and it will execute the task (perform the test). This is significantly different than the system of Kraft, and is not taught or suggested by Kraft. Nor does Silva, nor does the Examiner allege any such teaching in Silva. As none of the cited references teach or suggest this limitation of claim 1, alone or in combination, or the similar limitations of claims 8 and 15, the rejections of claims 1-3, 8-10, and 15-17 are legally and factually deficient.

The Examiner responds in the Advisory Action that there is no requirement of *when* various communications are sent, and that the claims only indicate an "eventual testing" that is confirmed when the test results are returns. This is a misreading of the plain language of the claims. The claims require a response that indicates that the client system *will perform* a test, clearly prospective language, and indicating that the client system *was not being actively used* when the executable program code was sent. The response clearly is sent after the program code is sent to the client, but before the test is performed.

Claim 4 requires receiving executable code from a server system in a client data processing system. The Examiner again refers to Kraft's col. 9, lines 1-27, reproduced above. Kraft does teach that a "subtask" is received by peripheral computer 106. Though not specified, it may be from coordinating computer 102.

Claim 4 also requires if the client data processing system is in an idle state when the executable code is received, then sending a response to the server system, testing at least a portion of the executable code, and sending test results to the server system. The Examiner again refers to Kraft's col. 9, but Kraft does not teach or suggest this limitation. There is no response sent to the coordinating computer upon receipt of the subtask, whether or not the peripheral computer is idle when the subtask is received. Nor does Silva teach this limitation.

Claim 5 requires, in contrast, that if the client data processing system is not in an idle state when the executable code is received, then no response is sent to the server and no testing is performed. This is also contrary to Kraft, which does not address the state of the client when the subtask is received, but will certainly execute the task and send a result whenever it does become

idle. None of the cited references teach or suggest this limitation of claim 4, alone or in combination, or the similar limitations of claims 11 and 18. The rejections, therefore, of claims 4-7, 11-14, and 18-21 are also legally and factually deficent.

CONCLUSION

As a result of the foregoing, the Applicant asserts that the claims in the Application are in condition for allowance over all art of record, and that the rejections are both factually and legally deficient, and respectfully requests this case be returned to the Examiner for allowance or, alternatively, further examination.

The Commissioner is hereby authorized to charge any additional fees connected with this communication or credit any overpayment to Munck Carter Deposit Account No. 50-0208.

Respectfully submitted,

MUNCK CARTER, P.C.

Date: 7/21/3

Matthew S. Anderson
Registration No. 39,093

P.O. Drawer 800889
Dallas, Texas 75380
(972) 628-3600 (main number)
(972) 628-3616 (fax)
E-mail: *manderson@munckcarter.com*